



(19)

Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 0 822 679 A2

(12)

## EUROPEAN PATENT APPLICATION

(43) Date of publication:  
04.02.1998 Bulletin 1998/06(51) Int. Cl.<sup>6</sup>: H04J 13/00

(21) Application number: 97108933.9

(22) Date of filing: 03.06.1997

(84) Designated Contracting States:  
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC  
NL PT SE

(30) Priority: 03.06.1996 DE 6507250

(71) Applicants:

- HE HOLDINGS, INC. dba HUGHES ELECTRONICS  
Los Angeles, CA 90045-0066 (US)
- City University of New York  
New York, New York 10036 (US)

(72) Inventor:  
The designation of the inventor has not yet been filed(74) Representative:  
Otten, Hajo, Dr.-Ing. et al  
Witte, Weller, Gahlert, Otten & Stell,  
Patentanwälte,  
Rotebühlstrasse 121  
70178 Stuttgart (DE)

## (54) New CDMA code generator employing mixing ergodic transformation

(57) A codeword generator (12) produces codewords in the form of digital electronic signals which may be employed, for example, in spread spectrum communications systems or RADAR systems. The generator (12) effects a mixing ergodic transformation (24) upon a suitable irrational number to produce a complex number sequence. The generator truncates (30) the complex number sequence to a predetermined length, thus

forming a codeword. The generator then forms other codewords within a family by permuting (28) this first codeword. Other families of codewords may be generated by employing another mixing ergodic transformer (32) to transform the same irrational number or another suitable irrational number.

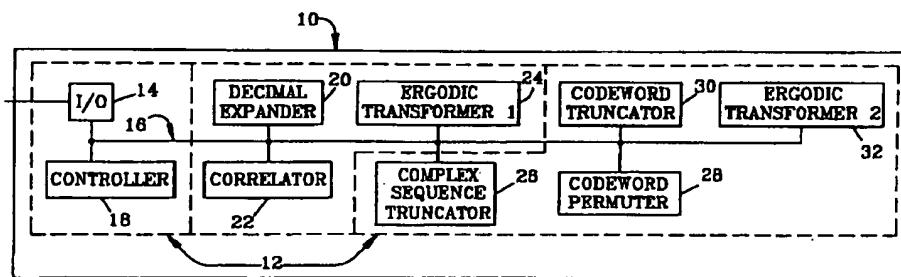


FIG.1

EP 0 822 679 A2

**Description****GOVERNMENT RIGHTS**

5       The government has certain rights in this invention pursuant to Contract Number N00014-94-C-0252 awarded by the Department of the Air Force to Hughes Aircraft Company and Contract Number F49620-93-1-0610 awarded to the City University of New York by the Department of the Air Force.

**BACKGROUND OF THE INVENTION**

10      **Field of the Invention**

The invention is related to pseudo-noise codes and, more particularly, to the generation of electronic signals which employ such codes.

15      **Description of the Related Art**

Spread-spectrum techniques are known and are employed in communications, guidance, and RADAR systems. See Raymond L. Pickholtz, Donald L. Schilling, and Laurance B. Millstein, Theory of Spread Spectrum Communications-A Tutorial, IEEE Transactions On Communications, VOL. COM.30, NO.5, May 1982, pp 855-884, for a discussion of the use of spread spectrum techniques in communications systems, for example. In a first order analysis, spread spectrum communications systems employ "pseudo noise" (PN) codewords at a transmitter to "spread" the bandwidth occupied by transmitted data beyond the minimum required by the data. The codewords are independent of the data being transmitted and are also employed by the system's receiver to synchronously receive and "despread" the transmitted data. Spread spectrum transmissions are less susceptible to jamming, to multipath interference and to fading than other transmissions.

Unlike time division multiple access (TDMA) communications systems, which permit multiple users to occupy the same bandwidth at different times, or frequency division multiple access (FDMA) systems, which permit multiple users to use different frequencies at the same time, spread spectrum code division multiple access (CDMA) techniques accommodate multiple users by permitting them to use the same frequency bandwidth at the same time. Ideally, each codeword yields a spectrum which appears as noise to other occupants of the frequency band and each additional user simply adds to the "noise floor" of the band until the maximum capacity of the channel is reached. Not only do CDMA communications systems provide security, i.e., the codeword is required to despread the signal, but, because the transmitted signals appear as noise to all but those who possess the codewords, it is difficult to even determine that a signal has been transmitted: an ideal attribute for covert communications systems.

RADAR systems also employ PN codes. For example, PN codes are used to modulate the RADAR's transmitted waveform so that multiple returns from different targets may be readily distinguished. Additionally, since the RADAR's pulses have been spread out in frequency and appear, to a first approximation, as noise, it is more difficult for a target to determine that it has been "lit up" by the transmitting RADAR.

In order to provide secure and covert communications and to modulate RADAR pulses with the greatest degree of flexibility possible, the PN codes, or codewords should be arbitrarily long and should provide an unlimited number of code families. That is, because the difficulty of "cracking" a code family increases significantly with the number of codewords available, and the number of codewords available within a family increases with the length of the codewords, the ideal code generator would provide an infinite number of codes, i.e., infinitely (arbitrarily) long codewords. Additionally, the codes should be nonpredictable, that is, a small sample set of the codes should not permit the prediction of subsequent codes.

To minimize errors, the codewords should be distinctive. That is, the crosscorrelation between any two codewords should be low and any codeword's autocorrelation should be steeply peaked, with small sidelobes. Minimal codeword crosscorrelation yields minimal interference between multiple users in a CDMA communications system or between multiple target reflections in a RADAR application. Similarly, maximally peaked codeword autocorrelation yields optimal acquisition and synchronization properties for both communications and RADAR applications. Furthermore, in order to avoid detection and acquisition, and to thereby enhance their utility in secure and covert applications, the codewords should exhibit additive white Gaussian noise (AWGN) properties. That is, the codeword power spectrum should resemble that of AWGN and the distribution of crosscorrelation values, both real and imaginary, should be Gaussian. In addition, to enhance target discrimination in RADAR applications, the power content of the code sidelobes (the average magnitude of the codewords' crosscorrelation values) should be inversely proportional to the number of codeword pairs that are averaged.

However, conventional codes, such as Gold codes and M sequences generally are not arbitrarily long. Existing

code generators employ shift registers to generate codes, such as Gold codes and M-sequence codes. Unfortunately, these codes are limited in length to  $2^N-1$ , where N is the number of locations within the generating shift register. Furthermore, existing codes do not yield AWGN spectral characteristics. For example, a histogram of Gold codes' auto- and cross-correlation values does not possess desirable, noise-like Gaussian distribution characteristics, rather, it has  
 5 three distinct values. Consequently, signals which employ these codes are more easily detected than may be desired. That is, a transmission from a covert communications system employing a Gold code would be more easily detected than a transmission which produces AWGN statistics because the transmissions with AWGN statistics appear as noise to an interceptor, whereas a non-AWGN transmission will exhibit characteristics that identify it as a transmission with some information content. Similarly, it would be much easier for a RADAR target to determine that it has been "lit up"  
 10 by RADAR employing codes which do not produce AWGN statistics.

With conventional codeword generators, the number of code families is limited and the number of codewords within each family is usually less than or equal to the codeword length. Furthermore, linear codewords built from shift registers can be predicted if N bits out of  $2N-1$  bits of the codeword are known. This precludes the use of such codes in secure and covert communications applications. Additionally, the distribution of crosscorrelation values for conventional codewords is not Gaussian, again limiting the utility of such codes in covert and secure communications applications and in  
 15 RADAR applications.

#### SUMMARY OF THE INVENTION

20 The invention is directed to a codeword generator and method of generation that reduces the detectability and improves the security of systems which employ the codes.

The new codeword generator produces codewords in the form of digital electronic signals by operating a mixing ergodic transformation upon a suitable irrational number to produce a complex number sequence and then truncating the sequence to a predetermined length to form a codeword. The generator then forms other codewords within a family  
 25 by permuting the first codeword. Other families of codewords may be generated by employing another mixing ergodic transform to transform the same irrational number or another suitable irrational number.

In a preferred embodiment, the desired codeword length and a decimal expansion of a suitable irrational number are provided to the generator "off-line". However, other embodiments of the generator may include an irrational number decimal expander and autocorrelator which respectively, expand and test for the suitability of candidate irrational numbers.  
 30

The new codeword generator is particularly suited to application in covert and secure communications and RADAR systems, where it may produce an unlimited number of codewords for combination with data or RADAR pulses and thereby thwart interception of these transmissions. Furthermore, the codewords exhibit the AWGN characteristics which make transmissions built upon them appear noise-like, further enhancing their utility in these applications. Additionally, the new codewords are nearly orthogonal to Gold code codewords and, therefore, may be operated along with  
 35 existing systems using Gold codes in the same channel with minimal interference. The new codeword generator provides an infinite number of code families and the codewords within each family may be arbitrarily long. Unlike conventional linear codes, codes produced by the new code generator are not predictable from a small sample set. The codewords are complex-valued, of unit magnitude, and may be implemented with varying numbers of discrete complex  
 40 values, providing a tradeoff between ease of implementation and AWGN statistics.

These and other features, aspects and advantages of the invention will be apparent to those skilled in the art from the following detailed description, taken together with the accompanying drawings.

#### BRIEF DESCRIPTION OF THE DRAWINGS

45 FIG.1 is a functional-level block diagram of a codeword generator system based upon the new codeword generator.  
 FIG.2 is a flow chart which illustrates the sequential operation of the new codeword generator.  
 FIG.3 is a block diagram of a preferred embodiment of the new codeword generator.  
 FIG.4 is a block diagram of a communications system that employs the new codeword generator.  
 50 FIG.5 is a flow chart that illustrates the operation of a communication system which employs the new codeword generator.  
 FIG.6 is a block diagram of a RADAR system which employs the new codeword generator.  
 FIG.7 is a flow chart depicting the use of the new codeword generator in the transmission and reception of a RADAR pulse.  
 55

#### DETAILED DESCRIPTION OF THE INVENTION

The new codeword generator produces codewords in the form of digital electronic signals that may be used, for

example, in spread spectrum communications and RADAR systems. The functional level block diagram of FIG. 1 illustrates the primary functions of a codeword system 10 based upon a new codeword generator 12. The system functions not contained within the codeword generator blocks 12 of this exemplary illustration could, optionally, be included within the codeword generator 12. An input/output system (I/O) 14 provides a bidirectional communications link to systems outside the codeword generator system 10. The I/O system 14 is connected through an internal bidirectional communications link 16 to the other components of the codeword generator system 10. The link 16 provides a communications path for all the components of the new code generator 12. A controller 18 is connected through the link 16 to the remaining system components and controls the timing and interaction of those components.

In the preferred embodiment, a decimal expander 20 and correlator 22, although a part of the code generator system, are "offline" from the codeword generator 12. The decimal expander, as will be described in greater detail in relation to FIG. 2, provides a decimal expansion of irrational numbers, e.g., 3.1415 is a five digit decimal expansion of  $\pi$ . The correlator 22, again, as will be described in greater detail in relation to FIG. 2, autocorrelates a complex valued sequence based upon the decimal expansion provided by the decimal expander 20. The correlator 22 determines the suitability of a particular decimal expansion for use as a "basis" for codewords based upon this autocorrelation. A first mixing ergodic transformer 24 transforms candidate decimal expansions to produce the complex valued sequence which is autocorrelated by the correlator 22. Mixing ergodic transformations are known. For a detailed discussion see Cornfeld, Fomin, Sinai, Ergodic Theory, Springer-Verlag, 1982, pp 3-137.

Given a complex sequence developed by the ergodic transformer 24 and checked for suitability by the correlator 22, a complex sequence truncator 26 truncates the sequence to produce a codeword C1. A codeword permuter 28 permutes the codeword C1 to produce additional codewords. A codeword truncator 30 truncates codewords, under direction of the controller 18, to a desired length. To generate additional families of codes, a second ergodic transformer 32, which performs a different transformation from that of the ergodic transformer 24, transforms the decimal expansion used to generate the first code family. Additional transformers may be employed to generate additional code families, as may additional irrational number decimal expansions.

The flow chart of FIG.2 illustrates the sequential operation of the new code generator system 10. Generation of a code starts with BEGIN 34 and proceeds to step 36. In step 36 a mixing ergodic transform  $T(x)$  is chosen. In the preferred embodiment, a suitable transform is provided to the code generator. The transform  $T$  maps a value  $x$  defined on the unit interval  $([0,1])$  to another value  $T(x)$ , also on the unit interval. An example of a mixing ergodic transform is given by:

$$30 \quad T(x) = (nx) - \text{int}(nx)$$

where:

35    n       = an integer greater than or equal to 2  
 int      = a function which returns the integer value of its argument

In other words, the transform  $T(x)$  multiplies a number  $x$  by an integer greater than 2, subtracts the integer value of this product from the product, thereby yielding the product's fractional value, i.e., all the terms to the right of the decimal point. The desired codeword length L is a system design parameter, i.e., the degree to which the spectrum is spread is proportional to the codeword length. The total desired number of codewords N is also a system design parameter; the difficulty of "cracking" a code increases significantly with the number of codewords and, in many systems each user is assigned a separate codeword. Both N and L are downloaded to the code generator 12 through the I/O system 14 in the preferred embodiment. Experiments indicate that the transform  $T(x)$  works well with  $n = 10$ .

45    The code generator system 10 then proceeds to step 38 where a suitable basis for the code is found. In the preferred embodiment, this step is performed offline by the correlator 22 and ergodic transformer 24. Qualitatively, a suitable basis provides a good digit distribution, that is, one that doesn't contain any long digit patterns which repeat periodically. To find such a basis, the decimal expander 20 provides a decimal expansion W of an irrational number such as  $\pi$  or  $\sqrt{2}$  to the ergodic transformer 24. The ergodic transformer forms a complex-valued sequence Y of length Q :

$$50 \quad Y = (a_0, a_1, \dots, a_{Q-1})$$

where:

$$55 \quad a_k = \exp[j2\pi T^{k+p}(W)]$$

and:

$W$  = candidate irrational number  
 $p$  = starting point within the decimal expansion of  $W$   
 $T^m$  indicates that the ergodic transform  $T$  should be iterated  $m$  times

5 Once the ergodic transformer 24 constructs the complex number sequence in this fashion, the correlator 22 autocorrelates the sequence  $Y$ :

$$R_{YY}(n) = \sum Y(k)Y^*(k+n)$$

10 where the summation is taken from  $k = 0$  to  $Q-1$ .

If the maximum magnitude of  $R_{YY}(n)$  for all  $n > 0$  except  $n=0$  is relatively small (i.e., less than  $4/\sqrt{Q}$ ), the number  $W_p$ , the irrational number which "starts" at the  $p$ th location of the decimal expansion of  $W$ , is a suitable choice for a basis from which a new code may be produced.

15 After a basis has been selected in this fashion, operation proceeds to step 40 where the code generator 12 produces the first codeword of a code. For a given target codeword length  $L$ , the complex sequence truncator 26 truncates the complex sequence  $a_k$  to a length  $N_c$ , where  $N_c$  is a prime number greater than or equal to the desired codeword length  $L$ . That is, a first codeword  $C_1$  is generated:

$$C_1 = (a_0, a_1, \dots, a_{N_c-1})$$

20 where:

$C_1$  = continuous-valued complex number sequence

25 Since a practical implementation of the code generator requires some "quantization" of the complex values, the continuous valued complex numbers are quantized in this step. For example, if eight discrete complex numbers are allocated as possible values for each number  $a_k$ , each  $a_k$  is assigned whichever of the following values,  $1, \sqrt{2}/2 + j\sqrt{2}/2, 0 + j, -\sqrt{2}/2 + j\sqrt{2}/2, -1, -\sqrt{2}/2 - j\sqrt{2}/2, -j$ , or  $\sqrt{2}/2 - j\sqrt{2}/2$  is closest, yielding the codeword:

$$30 C_1 = (a_0, a_1, \dots, a_{N_c-1})$$

The quantization numbers e.g.,  $1, \sqrt{2}/2 + j\sqrt{2}/2$ , may be "fixed" in permanent memory locations or they may be downloaded as a parameter.

35 Next, in step 42, the code generator produces additional codewords within the codeword permuter 28 by permuting the first codeword  $C_1$  as follows:

$$C_m(k) = C_1(\text{mod}((m)(k+1)), N_c) \text{ for } 2 \leq m \leq N_c-1, 0 \leq k < N_c$$

40 That is, the value of code  $C_m$  at index  $k$  equals the value of code  $C_1$  at index  $(m)(k+1)$ , modulo  $N_c$ . For example, if the code length  $N_c$  is 997, the value of code  $C_{113}$  at index 50 equals the value of code  $C_1$  at index  $(51)(113)$ , or 5736, modulo 997, which equals 778 or,  $C_{113}(50) = C_1(778)$ . The permuter 28 generates  $N_c-2$  additional codewords by "shuffling" the original codeword  $C_1$  in this manner.

45 After generating the additional codewords, the controller 18 determines in step 44 whether the desired codeword length  $L$  is less than the value of the prime number  $N_c$  and, if it is, the generator 12 proceeds to step 46 where the codeword truncator 30 truncates the codewords to length  $L$ . That is, the last  $N_c-L$  complex values of each codeword are deleted by the truncator 30. This step allows the construction of code families where each member, or codeword, has an arbitrary length from 1 to  $N_c$ , and  $N_c$  may be arbitrarily large. After truncating the codewords, the generator 12 proceeds to step 48, as it does directly if the controller 18 determines in step 44 that the desired codeword length equals the prime number  $N_c$ .

50 In step 48, the controller 18 determines whether the target number of codewords  $N$  has been generated and, if they have, the generator proceeds to step 50 END and terminates operation. If, on the other hand, more codewords are required, the generator proceeds to step 52 where another basis for additional codewords is generated. The second mixing ergodic transformer 32, which employs a mixing ergodic transformation that is different from the first transformer 24, transforms the original basis  $W_p$  to produce another basis  $W_r$ . That is, a new basis  $W_r$  is produced from the original basis  $W_p$ :

$$W_r = \phi'(W_p)$$

where:

$\phi^r$  = the second mixing ergodic transform

5 An example of another mixing ergodic transform  $\phi^r$  which may be used for this operation is given by the expression:

$$\phi(x) = (n)(x) + \beta - \text{int}[(n)(x) + \beta]$$

where:

10

$n$  = an integer

$\beta$  = a real number in the unit interval

Experiments indicate that the transform  $\phi^r$  works well with values of  $n = 2$  and  $\beta = 0$ . After generating an additional basis in this manner, the generator 12 returns to step 40 and proceeds as previously described.

The number of digits required in the decimal expansion of the initial basis  $W_p$  is determined by the number of codewords desired, the desired length of the codewords, the desired precision and the initial starting point  $p$ . Assume that for every complex value in the initial codeword  $C_1$  we require  $R$  digits of precision. Since the codewords are  $N_c$  digits long, there must be  $N_c + R$  digits in the expansion and, if the decimal expansion is begun at location  $p$ , there must be 20  $p + N_c + R$  digits for the first  $N_c - 1$  codewords. If  $N > N_c - 1$ , then additional bases must be generated and one additional digit is required for each new basis. The required number of digits in the decimal expansion of the initial basis is therefore given by the expression:

$$Q = p + R + N_c + \text{ceil}[N/(N_c - 1)] - 1$$

25

where:

30

$N_c$  = the closest prime number greater than the desired length  $L$

$R$  = the number of digits of precision required for phase quantization

35

$N$  = the total number codewords desired

$\text{ceil}$  = is a function which returns an integer that is equal to its argument or the next greater integer

For example, to generate 1990 codewords using the decimal expansion of  $\pi$ , with eight digits of precision and assuming  $p = 0$ , 996 codes may be generated from the first  $997 + 8 = 1005$  digits of  $\pi$ . An additional digit is required to generate a new basis for the second 996 codewords. The generation of the decimal expansion of irrational numbers is known. For example, the arithmetic geometric mean method for the decimal expansion of  $\pi$  is described in Borwein and Borwein,  $\pi$  and the AGM, pp 1-61, 219-229 Wiley Interscience, 1978. Other irrational numbers may be expanded using polynomial root-finding techniques such as Newton's method. See Forma S. Acton, Numerical Methods That Work, pp 178-203 Harper and Row, 1970.

40 Unlike conventional codes, the new code generator can generate more codewords in a family than the length of a codeword. For example, assume there is a requirement for a family of codes with each codeword 500 complex numbers long. The new generator 12 can use the prime number 997, generate 996 codewords, then truncate each of the codewords to 500 complex numbers in length. Even more codewords may be generated by starting with a larger prime number or by generating new bases as described above. The number of codewords in a family may also be increased 45 by changing the value of the  $p$ -digit offset. For example, assume the generator produces 996 codewords using the first 1005 digits of  $\pi$  (i.e.,  $p = 0$ ). The generator can then generate an additional 996 codes using the next 1005 digits of  $\pi$  by setting  $p = 1006$ . Further codeword generation may be achieved by changing the codeword family basis  $W$  to an expansion of  $\sqrt{2}$  or other irrational number, rather than  $\pi$ , for example. The codewords generated by the different bases retain orthogonality and the desirable code properties described above. Furthermore, longer codewords may be generated 50 by concatenating shorter codewords. For example, a codeword of length 9970 could be produced by concatenating ten codewords of 997 complex numbers.

55 A preferred embodiment of the code generator 12 includes the components illustrated in FIG.3. The I/O system 14, bidirectional data link 16, and controller 22 are as described in relation to FIG.1. An arithmetic and logic unit ALU 54 and digital storage 56 are also connected to the data link 16. The physical implementation of these components may take many forms. For example, an application specific integrated circuit (ASIC), or digital signal processing integrated circuit, DSP chip, could incorporate all the illustrated components. The storage 56 could be implemented as electronic, magnetic, or optical storage and, in the preferred embodiment such values as the total number of codewords desired  $N$ , the desired codeword length  $L$ , a prime number greater than the desired codeword length  $N_c$ , a decimal expansion

W<sub>p</sub>, the quantization numbers, and instructions required to perform a first ergodic transform T(x) and second ergodic transform φ(x) may all be contained within the storage system 56.

In the preferred embodiment, the decimal expansion is performed offline and downloaded to the generator's storage system 56 through the I/O interface 14. All the remaining operations required to produce codewords can readily be 5 performed in real-time. The generator employs a table look-up technique to generate the complex value sequence. The generator provides additional flexibility, in that the number of entries in the table determines the number of complex values used to quantize the numbers which compose the complex number sequence from which the codewords are generated. For example, the use of two discrete complex values (1,-1) for quantization, yields real-valued codewords with 10 limited AWGN properties. Complete AWGN characteristics are produced using only eight discrete complex values (listed above in reference to step 40). Switching between codewords generated with different initial bases W and digit offsets p may be accomplished by storing the different bases and performing the rest of the calculations in real time.

A spread spectrum communications system may employ the new code generator as illustrated in the block diagram of FIG.4. Digital data, which may include digitized voice, measurements, instructions, etc., is gathered in block 58 e.g., a multiplexer and passed to an encoder 60 which encodes the data for error detection and correction. After encoding, 15 the data is sent to a packetizer 62 which formats the encoded data, adding headers, etc. A multiplier 64 then multiplies the data with codes generated by the new code generator 12. This operation is known, for a more detailed discussion see Raymond L. Pickholtz, Donald L. Schilling, and Laurance B. Millstein, Theory of Spread Spectrum Communications-A Tutorial, IEEE Transactions On Communications, VOL. COM.30, NO.5, May 1982, pp 855-884. The resulting digital signal is sent to the transmitter 66 where it is converted from digital to analog form, modulated, amplified, etc. 20 and passed to a transmitting antenna 68.

On the receiving end, an antenna 70 receives the signal transmitted by the antenna 68. A receiver 72 amplifies the received signal, demodulates it and converts it to digital form. The resulting digital data stream is sent to multipliers 74 and 76. A new code generator 12 provides codes to the multipliers 74 and 76 which are identical to those generated by the code generator on the transmit side of the communications system. A detect, acquire, and track system 78 determines 25 from the output of the multiplier 74 when a signal from the transmitter 66 has been received, acquires the signal by "locking on" to it and tracks the signal and, in so doing, provides timing information to the multiplier 76 which allows it to synchronize codes from the generator 12 with the incoming data stream and produce an output data stream from which data symbols may be detected, after depacketizing, error detection and error correction in the data processor 80.

The flow chart of FIG. 5 illustrates the operation of the new code generator 12 within a spread spectrum communications system. Operation begins at step 82 and proceeds to step 84, where data is gathered for transmission. In step 30 86 the gathered data is encoded for error detection and correction, then, in step 88, the encoded data is formatted for transmission. In step 90 the formatted data is multiplied with codewords from the new code generator 12 thereby spreading the spectrum occupied by the data. In step 92 the transmitter transmits the data which has thus been encoded. This step includes a number of substeps, including digital to analog conversion, modulation, amplification, 35 etc. that are known in the art.

In step 94 a receiver receives an electronic signal from a receiving antenna and, among other things, amplifies it with a low noise amplifier, demodulates the signal, and converts it from analog to digital form. The digital signal is then correlated in step 96 with identical codewords from a local codeword generator. In step 98 the transmitted data stream is detected, acquired and tracked using the results of the correlation of step 96. This provides synchronization for 40 another correlation to be performed in step 100, using the same codewords, which yields the transmitted data in step 100. Operation proceeds until all data has been transmitted and received, then to step 102 the end.

A RADAR system which employs the new code generator is illustrated in the block diagram of FIG. 6. A pulse generator 104 produces pulses which are multiplied in a multiplier 106 with codewords from the codeword generator 12. As noted in relation to FIG. 1, this process spreads the spectrum of the RADAR pulse, making it more difficult to detect by 45 those without access to the code generator 12. The RADAR pulse is converted to an analog signal, modulated and amplified in a transmitter 108 then transmitted, via an antenna 110 to a target. A reflected pulse arrives at the antenna 110 and is routed to a receiver 112, where it is amplified, demodulated, and converted from analog to digital form. The resulting digital signal is routed to a detection, acquisition and tracking system 114 which also receives codewords from the codeword generator 12. The codewords are used to correlate and thereby detect, acquire, and track the reflected 50 pulses. As is known in the art, the pulses are also Doppler processed and processed through "range gates". These processes yield information regarding the distance and relative speed of a reflecting target. Using various codewords enhances the discrimination of reflections from various targets.

The new code generator is employed within a RADAR system as illustrated in the flow chart of FIG.7. Operation begins in step 116 and proceeds to step 118 where a pulse is generated. In step 120, the pulse generated in step 118 is multiplied by a codeword generated by the new code generator 12. This code may be varied, depending upon the intended target, thereby easing the task of discriminating returned pulses. After multiplying the pulse by a codeword, the pulse thus modified is converted to analog form, processed, amplified and transmitted in step 122. After the pulse is reflected and received, it is amplified, processed and converted to digital form in step 124. The digitised return is then 55

correlated with codewords from the codeword generator to detect and acquire and track returning pulses in step 126. Additional processing, such as doppler processing and range determination is performed in step 128 and, from there, the operation proceeds to step 130, the end.

Codes generated by the new codeword generator are unlimited in either the length of codeword or in the number of codewords in a given family. Due to the random distribution of phases, an entire codeword is not predictable from just a portion of itself. This property is especially advantageous for secure communications applications. Although it is possible to predict a code family from an entire codeword, assuming that all the codewords in a family are generated from the same seed W and digit offset p, this predictability may be eliminated by creating code families consisting of codewords created from different values of W and p.

The forgoing description of specific embodiments of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise forms disclosed, and many modifications and variations are possible in light of the above teachings. For example, the quantizer could be implemented using the arithmetic logic unit. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application, to thereby enable others skilled in the art to best utilize the invention.

It is intended that the scope of the invention be limited only by the claims appended hereto.

### Claims

1. A codeword generator, comprising:

a first mixing ergodic transformer (24) connected to receive a decimal expansion of an irrational number and to transform said decimal expansion, thereby creating a first complex number sequence,

a truncator (30) connected to truncate said complex number sequence to a predetermined length thereby forming a first codeword, and

a permuter (28) connected to permute said truncated complex number sequence and to return permuted complex number sequences to said truncator, where said permuted sequences are truncated to produce a family of codewords.

2. The codeword generator of claim 1, characterized by:

a correlator (22) connected to autocorrelate said complex number sequence, and

a controller (18) connected to pass said complex number sequence to said truncator (30) if the results of said autocorrelation satisfy a predetermined standard.

3. The codeword generator of claim 2, characterized by:

a second mixing ergodic transformer (32) connected to receive said decimal expansion of an irrational number and to transform the decimal expansion, thereby creating a complex number sequence different from said first complex number sequence, said truncator (30) connected to receive and truncate said different complex number sequence and said permuter (28) connected to permute said truncated different complex number sequence and to return permuted sequences to said truncator (30) where said different permuted sequences are truncated to produce a different family of codewords.

4. A codeword generator for producing digital electronic signals that may be used, for example, as code division multiple access (CDMA) codes in spread spectrum communications systems, or as pulse-shaping codes in RADAR systems, comprising:

a complex sequence truncator (26) connected to truncate a complex number sequence and to thereby produce a first codeword,

a codeword permuter (28) connected to receive and permute said first codeword and to thereby produce additional codewords,

an input/output system (14) connectable to external circuitry to relay data and instructions between the external circuitry and said codeword generator (12).

a communications link (16) connecting said truncator (26), said codeword permuter (28), and said input/output system (14), and

5 a controller (18) connected through said communications link (16) to control the operation of said truncator (26), said codeword permuter (28), and said input/output system (14).

5. The codeword generator of claim 4, characterized by:

10 a mixing ergodic transformer (24) connected to transform a decimal expansion of an irrational number and to thereby generate and communicate said complex number sequence to said truncator (26).

6. A communications system, comprising:

15 a data stream generator (58),

16 the codeword generator (12) of any of claims 1-5, to produce a family of codewords,

20 a multiplier (64) connected to multiply said data by said codewords, and

25 a transmitter (66) connected to modulate, amplify, and transmit the resultant of said multiplication.

7. The communications system of claim 6, further comprising:

25 a receiver (72) connected to receive, demodulate, and amplify signals,

30 the codeword generator (12) of any of claims 1-5, to produce said family of codewords,

35 a first correlator (74) connected to correlate said data by said codewords, to thereby detect, acquire, and track incoming data, and to provide a synchronization signal,

40 a second correlator (76) connected to correlate, under control of said synchronization signal, said codewords by said data and to thereby produce a de-spread received data stream.

8. A RADAR system, comprising:

35 a pulse generator (104) connected to produce RADAR pulses,

45 the codeword generator (12) of any of claims 1-5, to produce a family of codewords,

50 a multiplier (106) connected to multiply said pulses by codewords produced by said codeword generator (12), and

55 a transmitter (108) connected to transmit the resultant of said multiplication.

45 9. A method of generating codes, comprising the steps of:

(A) forming a sequence of complex numbers by ergodically transforming a decimal expansion of an irrational number (38),

50 (B) truncating the sequence formed in step A to a desired codeword length, thereby forming a first codeword (40),

(C) permuting the codeword formed in step B to produce additional codewords (42),

55 (D) truncating the codewords formed in step C to the desired codeword length (46), and

(E) producing a digital electronic signal corresponding to the value of said code.

10. A method of communicating data, comprising:

- (A-E) generating codewords according to the method of claim 9,
- 5 (F) determining whether a predetermined number of codewords has been generated (48) and, if not, employing a different ergodic transform from that used in step A to ergodically transform said sequence of complex numbers (52) and repeating steps B through D until sufficient codewords have been generated,
- 10 (G) multiplying digital data signals by said codewords (90),
- (H) modulating the signal produced in step G (92),
- (I) amplifying the modulated signal produced in step H, and
- 15 (J) coupling the amplified signal of step I to an antenna for transmission.

11. A method of transmitting and receiving a RADAR signal, comprising:

- (A-E) generating codewords according to the methods of claim 9,
- 20 (F) determining whether a predetermined number of codewords has been generated (48) and, if not, employing a different ergodic transform from that used in step A to ergodically transform said sequence of complex numbers (52) and repeating steps B through D until sufficient codewords have been generated,
- 25 (G) generating pulse signals (118),
- (H) encoding said pulses by multiplying said pulses by said codewords (120),
- (I) transmitting said encoded pulses (122),
- 30 (J) receiving radio frequency signals (124), and
- (K) cross correlating said received signals with said codewords to detect, acquire, and track reflected pulses (126).

35

40

45

50

55

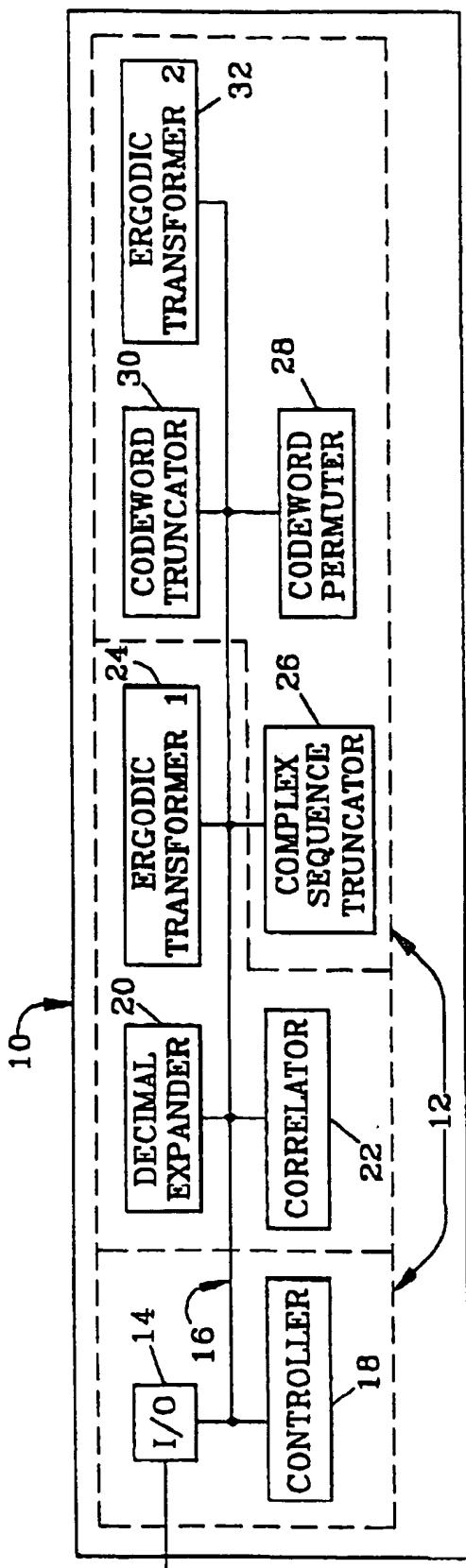


FIG. 1

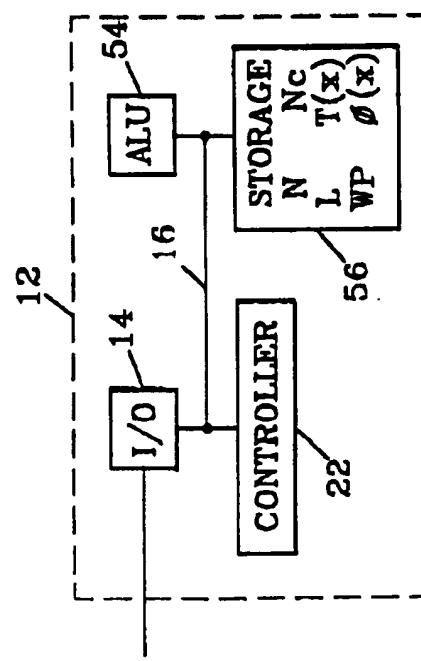


FIG. 3

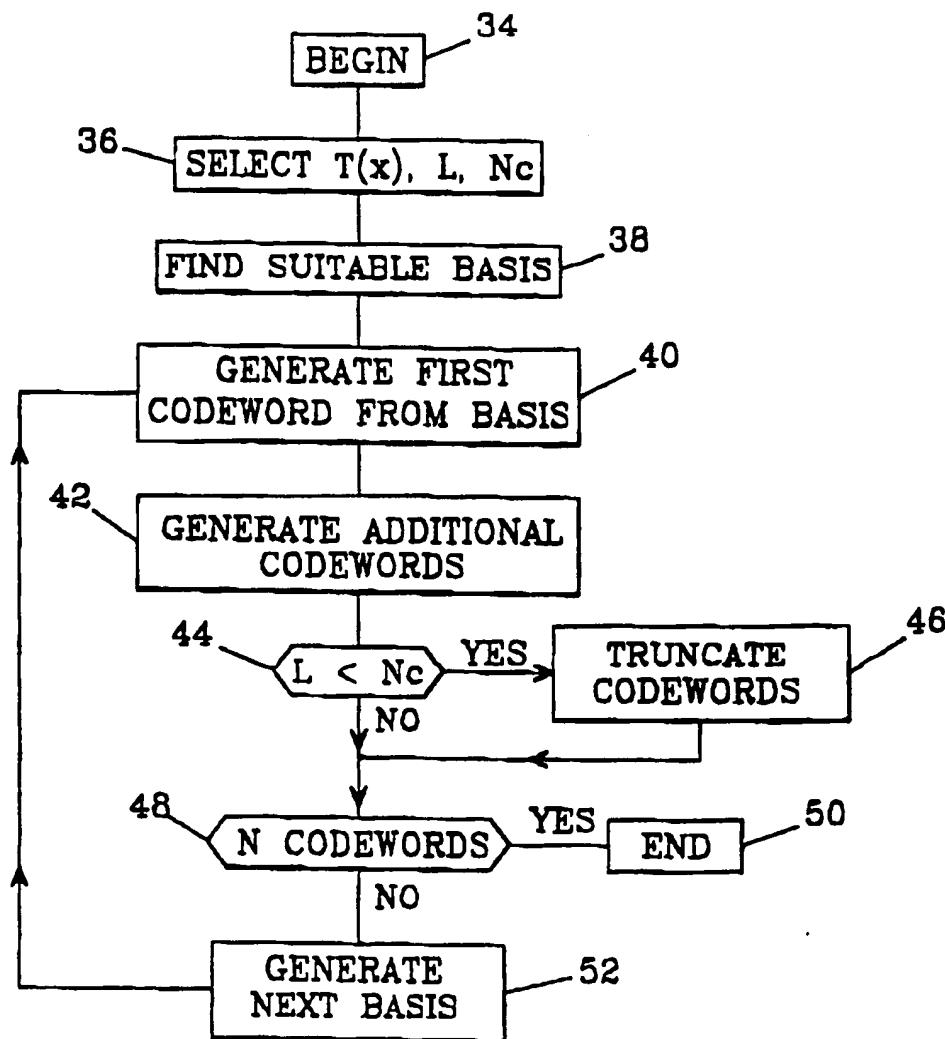


FIG.2

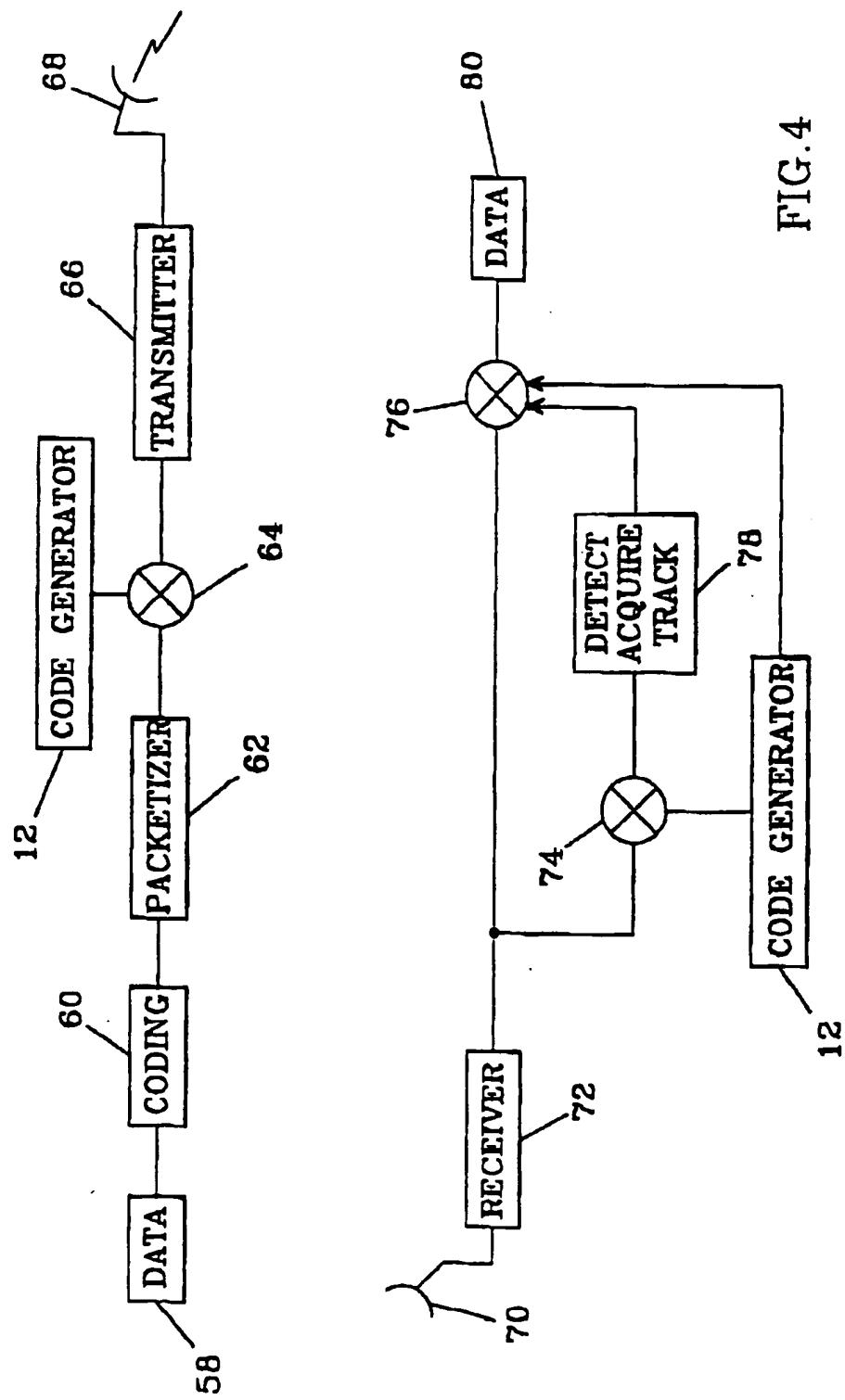


FIG. 4

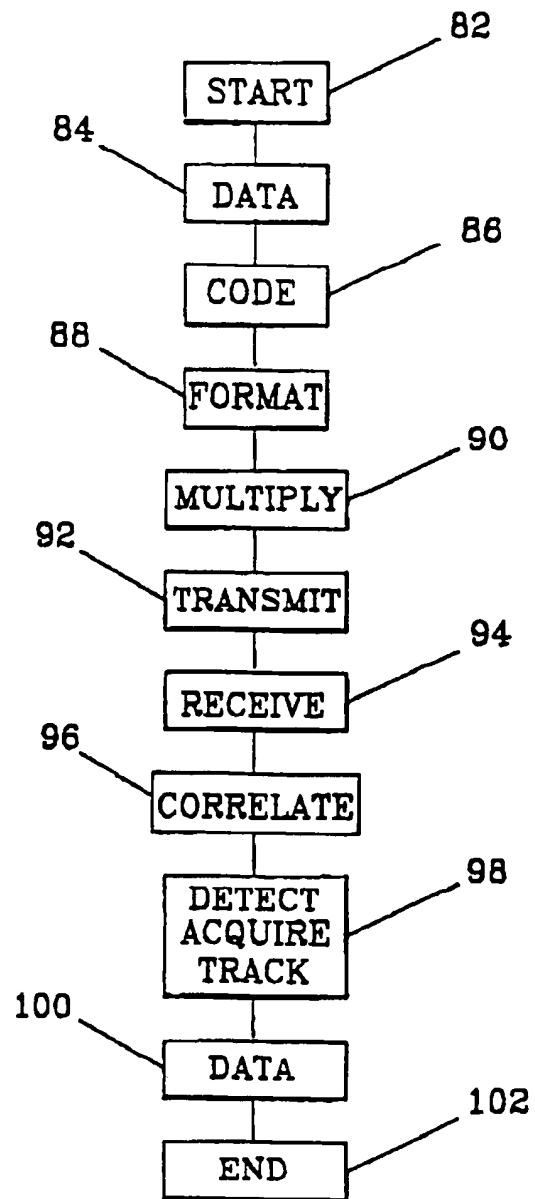


FIG.5

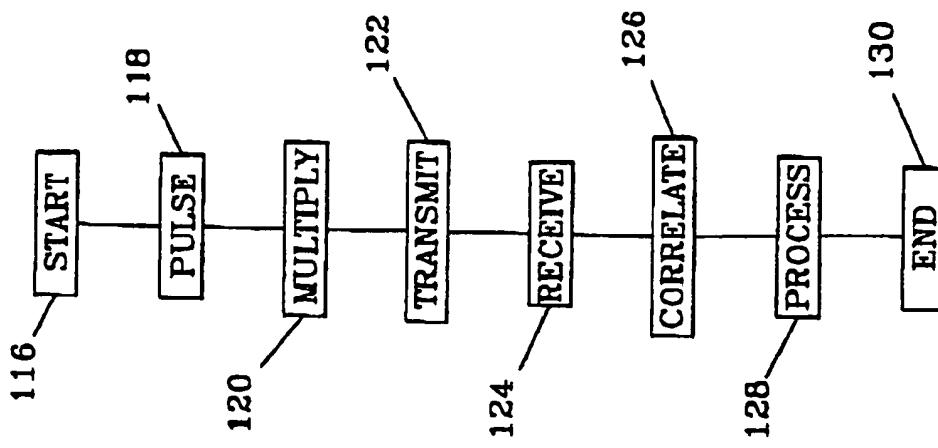


FIG. 7

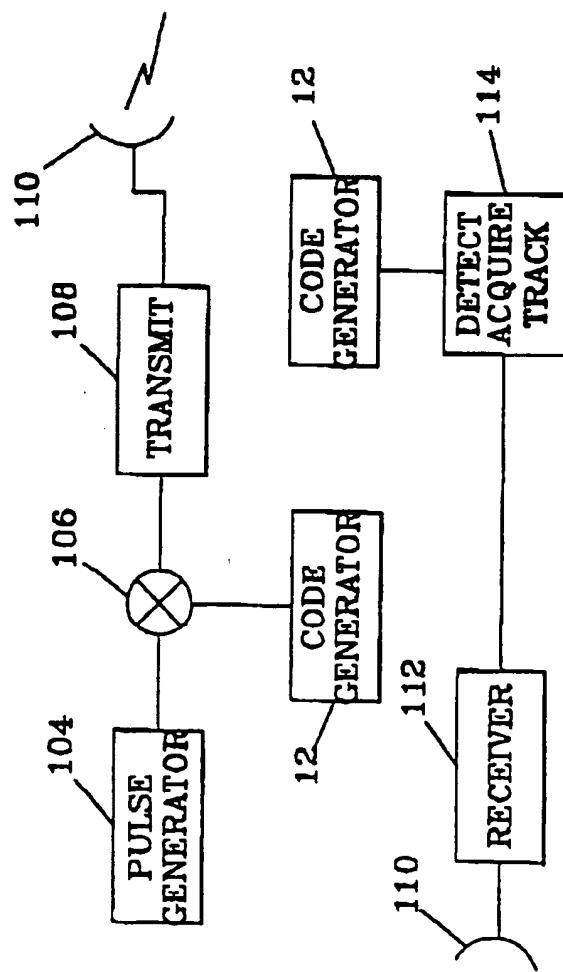


FIG. 6